Language Models can Generalize from Indirect Evidence: Evidence from Filtered Corpus Training (FICT)

Abhinav Patil

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2024

Committee
Shane Steinert-Threlkeld
Jaap Jumelet

Program Authorized to Offer Degree:
Linguistics

University of Washington

Abstract

Language Models can Generalize from Indirect Evidence:
Evidence from Filtered Corpus Training (FICT)

Abhinav Patil

Chair of the Supervisory Committee:
Shane Steinert-Threlkeld
Department of Linguistics

This thesis introduces **Fi**ltered **C**orpus **T**raining, a method that trains language models (LMs) on corpora with certain linguistic constructions filtered out from the training data, and uses it to measure the ability of LMs to perform linguistic generalization on the basis of indirect evidence. Applying the method to both LSTM and Transformer LMs, of roughly comparable size, we develop corpora filtered of direct evidence for a wide range of linguistic phenomena. Our results show that while transformers are better qua LMs (as measured by perplexity), both models perform equally and surprisingly well on linguistic generalization measures, suggesting that they are capable of generalizing from indirect evidence. This adds to a growing body of evidence on the limitations of perplexity as an evaluation metric, while also showing that direct attestation may be not strictly be necessary for learners to develop the appropriate linguistic generalizations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Table Number Page

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Shane Steinert-Threlkeld. His guidance and patient support have been invaluable throughout my time at the University of Washington and I feel immensely grateful to have been able to work with and learn from him. I am also deeply grateful to Jaap Jumelet for being an integral part of this project since its inception, for providing the graphic in Figure 1.1, and for agreeing to read my thesis.

Thanks should also go to those of my fellow UW students and alumni who made technical contributions and assisted with editing and proofreading throughout the lifetime of this project: Kelly Chiu, Andy Lapastora, Peter Shen, Lexie Wang, and Clevis Willrich. I am also grateful to the Linguistics Department and the members of the CLMBR lab, and to my professors and classmates here, whose support was essential to my being able to succeed and thrive. Thank you also to the UW Research Computing Club, the Hyak support staff, and Brandon Graves, for maintaining and providing assistance with the computing resources that were indispensable to the execution of this project.

Lastly, thank you to my family and friends, whose support and love has been the bedrock of my academic success.

Chapter 1

INTRODUCTION

Language models (LMs) play an increasingly large role in natural language processing systems and have become capable of producing surprisingly fluent and grammatical text. However, the mechanisms underlying the acquisition and use of such linguistic proficiency remain largely unknown. In particular, the degree that language learning relies on memorization versus generalization remains a topic of investigation (Hupkes et al., 2023). The reliance of LMs on large amounts of training data raises the suspicion that they do not generalize in a 'human-like manner' (McCoy et al., 2019; Hu et al., 2020; Oh and Schuler, 2023a), but it is hard to address such questions with traditional evaluation metrics such as perplexity. While traditional evaluation methods, such as perplexity over held-out corpora, provide valuable insights into a language model's predictive ability, multiple lines of evidence in recent research suggest that these measures may not fully capture the nuances of linguistic generalization in a human-like manner.

In response to these limitations, there is a growing interest in developing evaluation methodologies that target specific linguistic phenomena. One such approach, known as *targeted syntactic evaluation*, involves comparing a language model's preferences between pairs of sentences, where one is grammatical and the other is not (Marvin and Linzen, 2018). This methodology, exemplified by benchmarks like the Benchmark of Linguistic Minimal Pairs (BLiMP) (Warstadt et al., 2020), provides a standardized framework for assessing a model's ability to generalize linguistic rules.

This thesis introduces and explores the efficacy of **Fi**ltered **C**orpus **T**raining methodology (FICT) as a method for evaluating language model performance. As summarized in

Figure 1.1, the FICT methodology involves training models on corpora that have been filtered to remove specific linguistic environments, thereby testing the models' ability to generalize beyond their training data. By comparing models trained on filtered and unfiltered corpora, we hope to measure the extent to which language models can develop robust linguistic representations even in the absence of direct evidence.

We argue that the FICT methodology can potentially allow us to learn more about the inductive biases of a given model architecture. This is grounded in the hypothesis that language models with superior inductive biases should exhibit greater resilience to degradations in the quality of training data. We employ targeted filters to remove sentences containing specific linguistic phenomena from naturalistic corpora, focusing on areas identified by BLiMP benchmarks. Through extensive experimentation, we assess the impact of these filtered corpora on language model performance in syntactic and semantic domains. Thus, this thesis contributes to the body of literature on language model evaluation paradigms by providing empirical evidence for the methodological effectiveness of FICT. In particular, via this methodology, we demonstrate a dissociation between perplexity and the actual linguistic abilities shown by both LSTM and Transformer language models. Additionally, we show that while filtering has a negative impact on language model performance, this impact is overall quite small, which shows that learners can acquire appropriate linguistic generalizations even in the absence of direct evidence.

The work presented in this thesis is part of a broader investigation, the results of which have been prepared as a pre-print (Patil et al., 2024) which is currently under review at a venue. The methodology, as well as the core set of results, is shared between this dissertation and that pre-print. However, the pre-print includes an additional evaluation metric, called *probability delta*, which is not discussed here. Conversely, this thesis, but not the pre-print, includes a more detailed description of some of the filters used (§3.1), a regression analysis (§4.4), and an analysis of the relationship between corpus token count and our evaluation metrics (§4.1.2, §4.3.1).

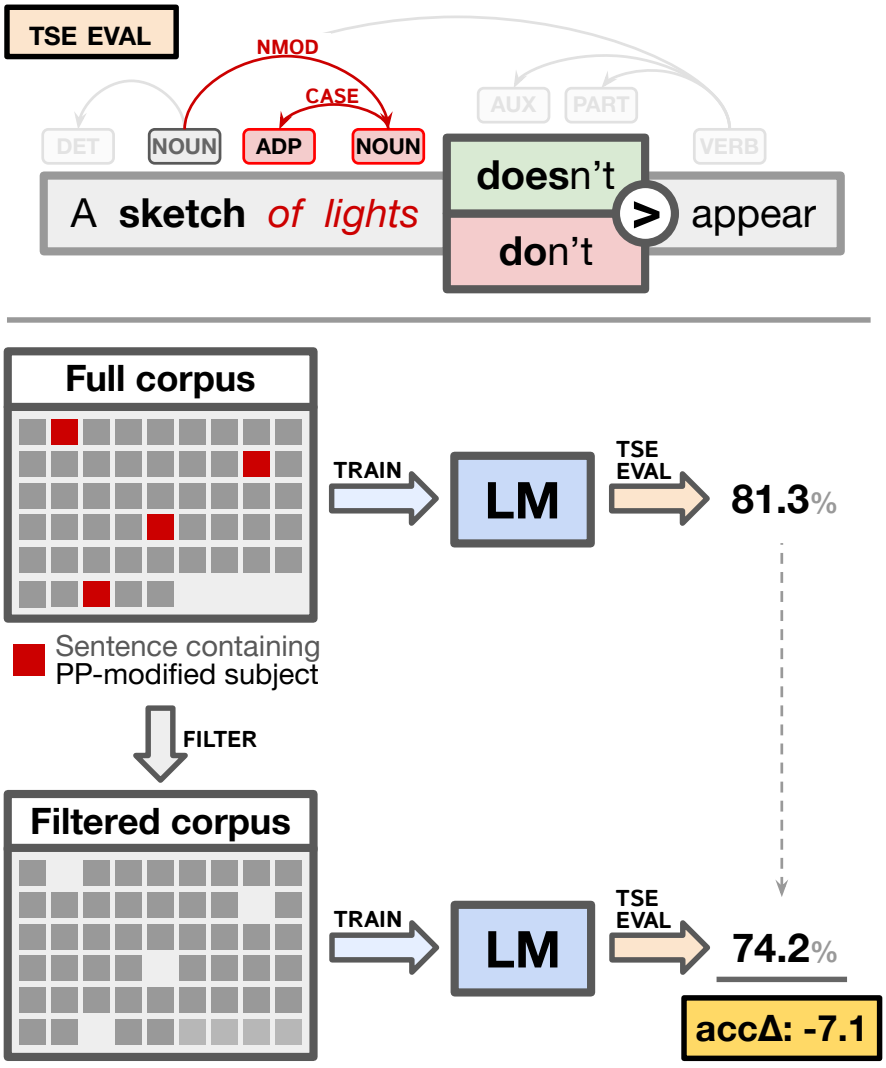

Figure 1.1: Overview of the FICT methodology. For a linguistic construction of interest (e.g. prepositionally modified subjects), we filter out sentences containing that construction and train a new language model on the filtered corpus. We measure performance on *targeted syntactic evaluations* to assess the capacity of the LM to generalize from related constructions to this novel, unseen construction.

Chapter 2

BACKGROUND AND RELATED WORK

2.1 Methods of Evaluating Language Models

2.1.1 Information-Theoretic Measures

By far the most commonly reported metric for language model performance is *perplexity* over a held-out corpus of test sentences. Perplexity is defined as

$$\text{PPL} = p(w_1, w_2, \cdots, w_N)^{-1/N} \tag{2.1}$$

where $w_0, w_1, \cdots, w_n$ are the words of the test corpus, and $\text{p}(w_0, w_1, \cdots, w_n)$ is the joint probability the language model assigns to them. However, it has been suggested that perplexity may fail to adequately measure the degree to which language models develop linguistic generalizations in a human-like way. Citing the famous sentence *Colorless green ideas sleep furiously* (Chomsky, 1957) as an example, Hu et al. (2020) points out that "a sentence can appear with vanishingly low probability but still be grammatically wellformed."

The literature presents us with evidence both for and against the idea that perplexity correlates with human-like linguistic generalizations and, more generally, for and against the utility of perplexity as an evaluation metric. For example, a body of psycholinguistic research has investigated the relationship between perplexity and human reading time. This literature is predicated on a hypothesis that certain human psychometric or behavioral measures, such as reading time, should correlate with information-theoretic metrics (such as perplexity) as measured by a probabilistic language model. Indeed, a number of papers (Frank and Bod, 2011; Fossum and Levy, 2012; Goodkind and Bicknell, 2018; Wilcox et al., 2020) find evidence of exactly such an effect. Conversely, however, Hao et al. (2020) investigated

this relationship for a much larger variety of models than prior papers—including LSTMs and transformers—and found evidence that this relation does not always hold. Similarly, Kuribayashi et al. (2021) found that, when using Japanese data, the relationship between perplexity and reading times attested in previous research on English-language data disappears, suggesting that these previous results may not generalize cross-linguistically. Curiously, Oh and Schuler (2023b) found that the relationship between information-theoretic performance for transformers and human reading time is strongest for models trained on two billion tokens or fewer, with the relationship progressively attenuating for transformers trained on a larger number of tokens.

2.1.2 Targeted Syntactic Evaluations

This paper builds on a growing body of literature that argues that information-theoretic measures, while still valuable in assessing model performance, should be augmented with other evaluations that specifically target the models' ability to generalize in a human-like way. Frequently, these investigations draw on psycholinguistic paradigms, treating language models as linguistic subjects in order to learn what such models "know" about specific linguistic phenomena (Linzen et al., 2016; Wilcox et al., 2018; Gulordava et al., 2018; Jumelet and Hupkes, 2018). A common paradigm in this body of literature, usually referred to as "targeted syntactic evaluation" (Marvin and Linzen, 2018; Futrell et al., 2019; Warstadt et al., 2020; Hu et al., 2020), involves comparing language models' preferences between "minimal pairs" of sentences, where one sentence is grammatical and the other ungrammatical. The primary metric in such evaluations is accuracy, where for a given minimal pair consisting of grammatical sentence S_1 and ungrammatical sentence S_2, a response is scored *correct* just in case:

$$\mathrm{P}_{\mathrm{LM}}(S_1) > \mathrm{P}_{\mathrm{LM}}(S_2)$$

Early work in this area (Linzen et al., 2016; Marvin and Linzen, 2018; Wilcox et al.,

2018) often used bespoke sets of minimal pairs tailored to the specific research questions and linguistic phenomena of interest. However, there are now a number of broad coverage benchmark sets of such minimal pairs, permitting standardized comparisons of language model performance. The benchmark with perhaps the widest coverage over linguistic phenomena is the Benchmark of Linguistic Minimal Pairs (BliMP, Warstadt et al., 2020). BLiMP consists of 67 different benchmarks, each consisting of 1,000 minimal pairs, which target twelve different linguistic areas, broadly construed, across morphology, syntax, semantics, and the syntax-semantics interface. This is the benchmark we use as a primary means of evaluation in the present investigation, as discussed in greater detail in §3.5.

2.2 Linguistic Generalization

While targeted syntactic evaluations give an insight into a model's linguistic competence, they do not show *how* a model acquires this notion of grammaticality. In this paper we focus on two kinds of linguistic generalization. First, *structural generalization* (Hupkes et al., 2023) asks: can language models make grammaticality judgments in syntactically more complex constructions than seen during training? Another line of work approaches this question from a fine-tuning perspective: by fine-tuning a model on a particular set of constructions we can measure the impact that this has on other linguistic constructions (Prasad et al., 2019; Weber et al., 2024). Secondly, *lexical generalization* asks whether models can generalize a seen construction to new lexical items that it has not seen in that construction (Kim and Linzen, 2020).

In order to gain a *causal* perspective on how the training data influences model performance, we retrain models from scratch on corpora *filtered* of specific types of sentences. Similar approaches have been deployed in a number of earlier works. For example, Misra and Mahowald (2024) investigate rare adjective-noun constructions and manipulate training corpora to investigate how models acquire an understanding of rare constructions. The paper with by far the most similar methodology to the one presented here is the "controlled ablation study" of Warstadt (2022). In that study, the author uses a dependency parser to

filter out sentences that offer evidence *against* an (incorrect) linear generalization for a single grammatical phenomenon (English subject auxiliary inversion). The researcher's primary investigative question is psycholinguistic: specifically, he is interested assessing evidence for or against psycholinguistic "poverty of the stimulus"-type arguments. In some ways, the present investigation is a natural extension of the psycholinguistic goals of Warstadt (2022), in that whereas that paper looks at only one linguistic phenomenon, here we look at fifteen. However, unlike that investigation, our research questions here are not solely concerned with psycholinguistic questions, and our experiments are designed accordingly. For example, whereas that paper considers only RoBERTa-style transformer models (along with a 5-gram baseline), we train and evaluate both LSTMs and transformers. This both allows us to determine what, if any, significant differences exist between these two architectures with respect to the specific linguistic phenomena being analyzed, and it also serves as a sort of control, allowing us to potentially arrive at conclusions that transcend model architecture. An additional difference between that paper and the present investigation lies in the type of transformer that is trained. Whereas the RoBERTa-style transformers of Warstadt (2022) were trained via a masked language modeling objective, we evaluate causal language models trained on next-word prediction. This is especially relevant for our evaluation methods. Both this paper and Warstadt (2022) use the targeted syntactic evaluation paradigm as a primary means of evaluation, the calculation of which requires that a model be able to assign a probability to each of a pair of sentences. Causal language models, unlike masked language models, naturally lend themselves to such an evaluation, as the latter lack a well-formed notion of "sentence probability" as such. While the aformentioned paper sidesteps this issue by using the "pseudo-log-likelihood" approximation of Salazar et al. (2020), by training causal language models, we avoid the need to use any sort of approximation.

As part of their investigation, Jumelet et al. (2021) also use a very similar methodology. This paper investigated the knowledge of recurrent language models with regards to a semantic phenomenon: negative polarity items (NPIs). After conducting a series of probing experiments using diagnostic classifiers (Hupkes and Zuidema, 2018) to determine whether

language models build linguistic representations that capture the notion of "monotonicity," an attribute of linguistic environments that largely governs the acceptability of various NPI constructions, the authors conduct two experiments that leverage a corpus filtering technique. In the first such experiment (the fourth out of all experiments in the paper), the authors train new models on corpora that have been ablated of various sorts of NPIs to see if this affects the aforementioned representations of "monotonicity." From this experiment, the researchers concluded that recurrent models are "still able to build up a shared robust notion of monotonicity" even in the total absence of NPIs in the training data. In the last experiment of the paper, the authors investigate whether recurrent language models can generalize the relationship between NPI's and monotonicity to environments in which they have never before seen them. To this end, they create nine corpora, each of which have been filtered of sentences with NPIs in a specific NPI-licensing environment. They find that recurrent models are generally able to make this sort of generalization, providing evidence against the idea that they are reliant on simple collocational clues and in favor of the hypothesis that the models are truly developing linguistic generalizations in a human-like way. The present study directly builds on this paper, and in particular, on the final experiment. In fact, as discussed in chapter 3, the three training corpora relating to negative polarity items used in this paper (NPI-ONLY, NPI-SENT-NEG, and NPI-SIM-QUES; see Table 3.1) are simply downsampled versions of three of the nine filtered corpora produced in that paper. However, whereas the investigators of that paper focused only on recurrent language models, here we also train and evaluate transformers, and whereas they focused only on NPI-related phenomena, our work covers twelve linguistic phenomena across syntax and semantics, corresponding to 31 of the 67 benchmark sets in BLiMP.

More generally, the literature contains many examples of researchers experimentally manipulating corpora in order to control for some potential confound. For example Wei et al. (2021), while not employing a *filtering* methodology as such, experimentally manipulate the frequency of specific "verbs of interest" across their training corpora, as these frequencies are potential confounds for their main experimental question. Similarly, in one of the ex-

periments conducted in Linzen et al. (2016), the researchers train models on corpora where the sentences have been stripped of all words that are not nouns, leaving only the original nouns in their original order, and then evaluate these models' ability to predict the number of a follow verb. Like Wei et al. (2021), however, they use these models only as baselines to control for syntactic information carried by non-nouns; unlike the outputs of our filters, the resulting corpora do not consist of grammatical English sentences. The present investigation differs from these and similar studies in that it manipulates the training corpora not just to establish a baseline or control for some latent variable, but specifically as a means of assessing the ability of learner to generalize grammatical rules to novel, unseen environments.

Chapter 3
METHODOLOGY

In this chapter, we discuss the methodology of the present investigation. First, we provide a detailed presentation of the implementation and design logic of each filter we used. Then we discuss the architectures and implementatitons of the models we trained and their training regimes. Note that code and data, as well as a link to all models on the HuggingFace Hub, can be found at https://github.com/CLMBRs/corpus-filtering.

3.1 Filters

The filters we used are listed in Table 3.1, along with the BliMP benchmark(s) each targets, and some descriptive summary statistics for each. As previously alluded to in §2.2, our filters target both structural and lexical generalization. Apart from the three NPI-related filters, which employ a simple regular expression matching heuristic to identify the relevant linguistic environments and remove sentences that contain them, the remaining twelve filters operate via custom logic implemented in the Python programming language. These filters utilized part-of-speech, morphological feature, and syntactic dependency annotations generated via the use of Stanza (Qi et al., 2020), an off-the-shelf package that uses pretrained neural models to generate grammatical annotations within the framework of Universal Dependencies (UD) (Nivre et al., 2017, 2020),[1] an open community project to formulate a set of standardized, consistent, crosslingual terminology and guidelines for the annotation of grammar. UD annotations fall into three categories:

1. **Universal POS tags**, categorizing tokens into broad part-of-speech categories.

[1] https://universaldependencies.org/

Corpus name	BLiMP benchmark	Proportion of benchmark filtered out	Pre-downsampling			Post-downsampling	
			Lines	Tokens	% lines filtered out	Tokens	Tokens as % of full
FULL	N/A	N/A	3052726	83058298	0.00	66442068	100.00
AGR-PP-MOD	distractor_agreement_relational_noun	0.995	2488090	64854017	18.50	63650020	95.80
AGR-REL-CL	distractor_agreement_relative_clause	0.944	2968459	79954914	2.76	65769189	98.99
AGR-RE-IRR-SV	irregular_plural_subject_verb_agreement_1	0.994	2708019	72645823	11.29	65507875	98.59
	irregular_plural_subject_verb_agreement_2	0.972					
	regular_plural_subject_verb_agreement_1	0.993					
	regular_plural_subject_verb_agreement_2	0.991					
NPI-ONLY	only_npi_licensor_present	1.0	3049998	82943806	0.09	66396500	99.93
	only_npi_scope	1.0					
NPI-SENT-NEG	sentential_negation_npi_licensor_present	1.0	3039128	82537646	0.45	66325213	99.82
	sentential_negation_npi_scope	1.0					
NPI-SIM-QUES	matrix_question_npi_licensor_present	1.0	3052276	83042884	0.01	66430070	99.98
QUANTIFIER-SUPERLATIVE	superlative_quantifiers_1	0.985	2830115	75246457	7.29	64929456	97.72
	superlative_quantifiers_2	0.993					
QUANTIFIER-EXISTENTIAL-THERE	existential_there_quantifiers_1	0.991	3017716	81956751	1.15	66325730	99.82
BINDING-C-COMMAND	principle_A_c_command	0.966	3052468	83048213	0.01	66439549	100.00
BINDING-CASE	principle_A_case_1	1.0	3005834	81405567	1.54	66135617	99.54
	principle_A_case_2	0.925					
BINDING-DOMAIN	principle_A_domain_1	1.0	3039381	82574625	0.44	66336878	99.84
	principle_A_domain_2	0.993					
	principle_A_domain_3	0.995					
BINDING-RECONSTRUCTION	principle_A_reconstruction	0.991	3052555	83052088	0.01	66432796	99.99
PASSIVE	passive_1	0.969	2971160	80491556	2.67	66155000	99.57
	passive_2	0.989					
DET-ADJ-NOUN	determiner_noun_agreement_with_adjective_1	0.956	3017929	81923257	1.14	66292740	99.78
	determiner_noun_agreement_with_adj_2	0.93					
	determiner_noun_agreement_with_adj_irregular_1	0.92					
	determiner_noun_agreement_with_adj_irregular_2	0.939					
DET-NOUN	determiner_noun_agreement_1	0.997	3038326	82607944	0.47	66406785	99.95
	determiner_noun_agreement_2	0.998					
	determiner_noun_agreement_irregular_1	1.0					
	determiner_noun_agreement_irregular_2	1.0					

Table 3.1: The training corpora used, along with some summary statistics for each. All the final corpora used in model training were downsampled to a uniform 2442181 lines. The columns under "pre-downsampling" refer to the corpora after filtering but prior to downsampling, while the columns under "post-downsampling" refer to the final corpora actually used in training.

2. **Universal features**, which differentiate tokens on the basis of additional lexical and inflectional characteristics (such as grammatical mood, tense, case, definiteness, etc.).

3. **Universal dependency relations**, syntactic relations that permit sentences to be expressed as a directed, weakly connected, edge-labeled, rooted graph, subject to the constraints that the single root node have no incoming arcs while all other vertices have exactly one incoming arc and that there exist a unique path from every vertex to the root; such a graph is commonly referred to as a *dependency parse* (Jurafsky and

Martin, 2024, chap. 18).

Some filters drew on all three levels of annotation, while others only relied on one or two. In the remainder of this section, we first discuss general design considerations underpinning the construction of all filters. Subsequently, we detail the specifics of the implementations of all filters, including the specific linguistic rationale informing their construction.

3.1.1 Design Considerations

By comparing models trained on the ablated data and models trained on the full, naturalistic corpus, we can potentially determine if, how, and when language models are able to make such generalizations. However, in order for this comparison to be maximally relevant, the locus of this comparison must be situated in a metric that is directly relevant to the sort of linguistic generalization that is informing which sentences are removed from the training corpora. For this, we turn to the targeted syntactic evaluations previously introduced in §2.1.2. Specifically, we design each filter to target the same phenomenon as one or more BLiMP benchmarks. Thus, we should expect that, minimally, when the filter is applied to the acceptable sentences in the BLiMP benchmark it targets, it should throw out 100% of them. We check whether this is true, the results of which are enumerated in Table 3.1. As can be seen in that table, about a third (9) of the 31 benchmarks meet this high standard, being filtered out by the corresponding filter with 100% accuracy. The others, however, generally came very close. The "weakest" filter by this metric (DET-ADJ-NOUN), still succeeded in removing 92% or more of all benchmarks which it targeted, with most other filters removing 95% or more of the benchmarks they targeted. Error analysis revealed that the vast majority of these errors were due to sporadic errors in the annotations generated by Stanza, with the remainder generally being the result of genuine ambiguity between multiple valid annotations.

The type I and type II error rate of each filter—and the trade-off between them—was a core design consideration. This trade-off is centered around two competing impulses: the first, to remove instances of a specific environment that attests a specific linguistic general-

ization so that the FICT methodology can actually work; and the second, to avoid removing so much information from the data that the model cannot possibly learn the generalization in question. On balance, however, we prefer a filter that minimizes type I error (a *stronger* filter), potentially at the cost of higher type II error. This is because, as applied in this investigation, the FICT methodology is fundamentally used to determine whether a language model can extend a linguistic generalization from seen to unseen environments. Thus, it is critical to the validity of such a methodology that the supposedly unseen environments *actually are unseen*. Consequently, all else equal, it is better for the filter to generate a false positive (filter out a sentence which actually does not contain the environment of interest) than to generate a false negative (permit a sentence to remain in the final corpus which actually does contain the environment of interest).

3.1.2 Filter Implementation

AGR-PP-MOD *and* AGR-REL-CL

The benchmarks targeted by both of these filters test subject-verb number agreement in the presence of an intervening *distractor* (in the terminology of Warstadt et al., 2020), defined as nouns which appear between a subject and its verb and have the opposite number from that subject. For the benchmarks targeted by AGR-PP-MOD and AGR-REL-CL, that distractor is realized in a prepositional phrase or in a relative clause, respectively. Thus, these benchmarks distinguish between models that develop an incorrect linear-ordering based account of subject-verb agreement in English, and ones that correctly develop a hierarchical or structural rule. An example of the minimal pairs in the benchmarks that these filters target follows, with the *distractor* italicized and the relevant **verb** in bold:

AGR-PP-MOD	A sketch *of lights* **doesn't** appear	A sketch *of lights* **don't** appear.
AGR-REL-CL	Boys *that aren't disturbing Natalie* **suffer**	Boys *that aren't disturbing Natalie* **suffers**.

The implementation of both these filters is effectively identical, and both target a type of structural generalization. Both remove sentences where the corresponding distractor appears between a noun and its verb, leveraging a dependency parse of the sentence to make this determination. As the original naturalistic training corpus surely contains other distractors between a noun and its verb, so will the corpora produced by these filters.[2] Furthermore, these filtered corpora will still contain prepositional phrases and relative clauses following nominals in other contexts. Thus, these filters test the ability of learners to develop a hierarchical account of subject-verb agreement (and syntactic structure, more generally) that can be applied even to environments where phrase structures appear in heretofore-unseen linear order.

AGR-RE-IRR-SV

The four BliMP benchmarks targeted by AGR-RE-IRR-SV all test language model performance on subject verb agreement, but two of them target "regular" plurals, like *dress/dresses*, while the other two target "irregular" plurals, like *goose/geese*. However, because we used the simple whitespace tokenizer of Gulordava et al. (2018) (as discussed in §3.2), our models lack access to subword information and thus have no notion of morphological regularity or irregularity. For our models, both regular *dress/dresses* and irregular *goose/geese* are each indivisible atoms, with no inherent relation. A successful learner in this context can be expected to develop linguistic representations for each pair of these tokens that are highly similar but for some notion of grammatical number, but there is no inherent reason a model should find such a pair of linguistic representations harder to develop for an irregular noun vs. a regular noun, or vice versa. Thus, our filter cannot be predicated on any notion of morphological regularity. As a result, this filter targets a sort of lexical generalization, rather than a structural one like the previous two filters.

The AGR-RE-IRR-SV filter operates by removing all sentences where a noun that appears

[2]Minimally, AGR-PP-MOD will contain relative clause distractors, and vice versa, though of course other distractors may occur in English.

in subject position is also used as a subject in these four benchmarks. The filter uses a dependency parse of the sentence to locate any nominal subject(s) and then cross-references this with a list of the nouns used in the benchmarks. Thus, in order to score well on these benchmarks, a model trained on the AGR-RE-IRR-SV corpus cannot rely on simple co-occurrence statistics, since it will have never seen a noun in the benchmarks in subject position. Instead, it must develop a notion of grammatical number, develop a linguistic representation of the grammatical number of the noun tokens in the benchmark based on their usage in other contexts, and then generalize the subject-verb agreement it sees for other nouns to these nouns.

NPI- *filters*

The three NPI-related filters are NPI-ONLY, NPI-SENT-NEG, and NPI-SIM-QUES; all three target a different kind of NPI-related structural generalization. These are simply downsampled versions of the Full\ONLY∩NPI, Full\S-NEG∩NPI, and Full\SMP-Q∩NPI corpora from that paper, respectively. As described in that paper, these filters use a simple regular expression matching heuristic to remove sentences: NPI-ONLY removes all sentences with an NPI occurring after 'only' (e.g. "Only students have ever complained about morning classes"), NPI-SENT-NEG removes sentences with a negation and an NPI, and NPI-SIM-QUES removes questions with NPIs in them.

QUANTIFIER-SUPERLATIVE

The two benchmarks associated with QUANTIFIER-SUPERLATIVE target the English grammatical rule that superlative quantifiers (e.g., at least, at most) cannot embed under negation. For example, the sentence *An actor arrived at at most six lakes* is grammatical, while **No actor arrived at at most six lakes* is not. The benchmark `superlative_quantifiers_1` consists of pairs where the grammatical and ungrammatical sentence both contain negation, but the the grammatical sentence has a comparative construction (e.g. more than, less than), while the ungrammatical sentence has a superlative. Conversely, `superlative_quantifiers_2`

consists of pairs where both sentences contain a superlative, but the ungrammatical sentence embeds this superlative under negation, while the grammatical sentence lacks such negation.

We note that the grammatical and ungrammatical sentences of both BliMP benchmarks contain a superlative or comparative in object position. The logic of the QUANTIFIER-SUPERLATIVE, which targets a structural generalization, is predicated on this observation: it uses morphological feature annotations to isolate comparative and superlative words and a dependency parse to determine if those words appear in object position, removing any sentence that contains a word meeting both criteria. Unlike all other benchmarks, it is much less clear *a priori* how a model trained on a dataset ablated in this way can develop a correct generalization. We observe that such constructions can appear outside object position in naturalistic English (c.f. in the contrasting grammaticality of *Not more than five forks are on the table. / *Not at least five forks are on the table.* and *At most six lakes contain fish. / *Not at most six lakes contain fish.*). However, such constructions are very rare. We also note that it has been suggested in the literature that such constructions are not licensed due to complex interactions between pragmatic and semantic effects. Both these facts serve as a potential, albeit tenuous, basis for a model to develop a correct generalization for this phenomenon.

QUANTIFIER-EXISTENTIAL-THERE

The benchmark associated with this filter, `existential_there_quantifiers_1`, targets the English rule that only so-called *weak* determiners/generalized quantifiers (e.g. many, at least, at most) are licensed in existential-*there* constructions, whereas *strong* determiners/generalized quantifiers (e.g. all, every, neither) are not (Milsark, 1974). Specifically, this benchmark uses the weak determiners *a, an, no, some, few, many* for its grammatical sentences, and the strong determiners *all, most, every, each* for their ungrammatical counterparts.

The QUANTIFIER-EXISTENTIAL-THERE filter removes sentences where any of these ten

quantifiers appear in an existential-*there* construction.[3] However, there are numerous other strong and weak determiners, and thus a model trained on this corpus will still see examples of this phenomena. In order to score well on these benchmarks, such a model cannot rely on simple co-occurrence statistics, since none of the determiners in the benchmark will have been seen in an existential-*there* construction. Rather, the model must develop a linguistic representation of the strength of these ten determiners based on their usage in other contexts, and then generalize the relationship it sees, for other determiners, between strength and licensing in existential-*there* constructions to determiners it has not seen in such a context. Thus, this filter targets a type of structural generalization.

BINDING- *filters*

Four filters, BINDING-C-COMMAND, BINDING-CASE, BINDING-DOMAIN, and BINDING-RECONSTRUCTION, target the seven binding-related benchmarks of BLiMP. All seven benchmarks typify various facets of Chomsky's (1993) Principle A. The implementations of all four filters, which each target a different binding-related structural generalization, is broadly similar: they target sentences where a reflexive or non-reflexive pronoun occurs in the specific context(s) illustrated by the corresponding benchmarks, narrowly construed, while leaving in sentences where the same or similar principle is applied in a different environment. For example, the ungrammatical sentences of the `principle_A_c_command` benchmark are ungrammatical because a reflexive pronoun and the noun it co-indexes are embedded in the same relative clause, and thus the anaphor fails to be c-commanded by its co-indexed noun; the grammatical counterparts of these sentence instead co-index the anaphor with the NP of the verb embedding the relative clause. Thus, the BINDING-C-COMMAND filter removes evidence of the use of the c-command relationship in anaphora licensing *in relative clauses*, but not elsewhere, as in sentences like *Mary's brother hurt himself* (but not **Mary's brother*

[3] In principle, the five strong determiners should never appear in such a position, but we target them anyways, in case this presupposition fails, thereby assuring that simple co-occurrence statistics will not lend themselves to a correct generalization.

hurt herself).[4] The other three benchmarks operate in similar ways.

PASSIVE

Two BLiMP benchmarks target the passive alternation in English. Both `passive_1` and `passive_2` contrast a sentence with a passivized transitive verb, which is grammatical and a passivized intransitive verb, which is not, but `passive_1` includes the thematic agent of the construction appears as a oblique dependent (embedded in a *by*-PP, e.g. Jeffrey's sons are insulted *by Tina's supervisor.*), whereas `passive_2` does not.

Much like AGR-RE-IRR-SV, the PASSIVE targets a lexical generalization and operates by removing sentences that contain words from a word list appearing a specific linguistic environment. Concretely, this word list consists of the verbs that are actually used in these two benchmarks in passive form, and the filter removes sentences where such words appear with the morphological annotation VOICE=PASS,[5] indicating they are in the passive voice.

DET-ADJ-NOUN *and* DET-NOUN

Like the benchmarks targeted by AGR-RE-IRR-SV, those targeted by DET-ADJ-NOUN and DET-NOUN are differentiated on morphological grounds, with two benchmarks each for nouns with regular plurals, and two each for nouns with irregular plurals. However, as noted in the discussion of the AGR-RE-IRR-SV, our models lack access to such morphological information, and our filters similarly do not make this distinction.

The DET-ADJ-NOUN removes sentences where an adjective appears between a demonstrative determiner (*this/these/that/those*) and its noun, acting as a distractor for agreement in a similar way as relative clauses and prepositional phrases do for the AGR-REL-CL and AGR-PP-MOD benchmarks, respectively. Much as for those filters, for a model trained on

[4]BLiMP assumes a straightforward one-to-one relationship between certain names and their grammatical gender. While such a relationship may not actually be borne out in practice today, the corpora used in this investigation likely do adhere to such a formulation.

[5]See `https://universaldependencies.org/u/feat/Voice.html#Pass`.

DET-ADJ-NOUN to succeed on the associated benchmarks, it must structurally generalize a hierarchical model of the relationship between determiners and their nouns from other, (non-demonstrative) determiners, and learn the grammatical number of these four demonstrative determiners from other contexts in which they appear. The latter of these two generalizations may be aided by the fact that, in English, the demonstrative pronouns are token-identical to the demonstrative determiners (c.f. *Give me those books* vs. *Give me those*).

The benchmarks associated with the DET-NOUN, on the other hand, do not involve adjectives, but rather contain sentences where a demonstrative determiner is immediately followed by its nominal argument. Much like the AGR-RE-IRR-SV and PASSIVE filters, the DET-NOUN targets a lexical generalization by removing sentences based on a word list. This word list consists of all nouns (in their singular and plural forms) that are actually used as the arguments of demonstrative determiners in the associated BLiMP benchmark sets. The filter then removes all sentences where such nouns appear as the arguments of the demonstratives *this/these/that/those*. Similar to models trained on the AGR-RE-IRR-SV corpus, a successful model trained on DET-NOUN must learn the grammatical number of the nouns in the noun list from other contexts. It must generalize numerical determiner-noun agreement from other nouns not on the list to those that are on it. Conversely, since these nouns do not ever appear with demonstrative determiners, a model that relies solely on co-occurrence statistics can be expected to perform poorly (approximately equal to chance, or 50%) on these benchmarks.

3.2 Data

The base training, validation, and testing corpora are the English language corpora used in Gulordava et al. (2018), as is the the vocabulary of the tokenizer used by all models. These data were created from dumps of English language Wikipedia. The sixteen training corpora used in this study were created via the follow process. The base training corpus was passed through each of the fifteen filters. The number of sentences and tokens discarded by each filter varied from as little as ∼0.1% to as much as ∼18.5%; for specifics, refer to the "pre-downsampling" columns of Table 3.1. Then, as an additional control, the fifteen filtered

corpora plus the original, full training corpus were uniformly downsampled to 2442181 lines, corresponding to ∼80% the size of the original training corpus. This downsampling was performed by discarding a randomly selected subset of the sentences of each corpus. It is worth noting that the number of *tokens* did vary by as much as ∼4.2%, as reflected in the rightmost column of Table 3.1.

3.3 Model Architectures

3.3.1 Architectures

Two architectures are used for the models trained in this investigation: recurrent neural networks—specifically, long-short term memory models (LSTMs, Hochreiter and Schmidhuber, 1997)—and Transformers (Vaswani et al., 2017). These two architectures were factorially combined with sixteen training corpora and five different starting seeds[6], resulting in a total of 160 models.

The LSTM models use a custom Python implementation written using the `torch` library (Paszke et al., 2019). The Transformer models use the `OPTForCausalLM` class of the HuggingFace `transformers` package (Wolf et al., 2020), which implements a causal decoder-only Transformer architecture like that of the General Purpose Transformer (GPT) of Radford et al. (2019) and the Open Pre-trained Transformer (OPT) of Zhang et al. (2022).

3.3.2 Hyperparameters

As an additional control, model hyperparameters were selected to equalize the total number of parameters between the LSTM and Transformer models as much as possible: the LSTMs and the Transformers had 68045650 and 67170816 trainable parameters, respectively. For the LSTMs, this meant a two-layer model with embedding and hidden dimension of 1024. Additionally, a dropout of 0.1 was used, and weights were tied between the embedding and output layers. The Transformers were constructed with feed-forward and hidden layer

[6] 0, 1, 2, 3, 4

dimensions of 768, eight attention heads, eight hidden layers, and 512 position embeddings. An identical dropout (0.1) was also used for all fully connected layers in the embeddings, encoder, and pooler, but attention dropout was set to 0.

3.4 Training

All models in this investigations utilize the same training regime. Model training was performed using the `Trainer` class of the `transformers` library. Models were trained on A40 GPUs for 40 epochs with mixed-precision training, using the AdamW optimization algorithm (Loshchilov and Hutter, 2017), with a linear scheduler, an initial learning rate of 5e−5, and a batch size of 32. A checkpoint was saved at the end of every epoch, and the loss on the validation corpus, measured for each of these checkpoints, was used for model selection. In all subsequent discussion in this paper, for each training run, all evaluations and results utilize the model checkpoint with the best perplexity on the validation corpus at the end of each of those 40 epochs. A complete set of training hyperparameters may be found in Appendix A.

3.5 Evaluation

We use three metrics as the primary means of evaluation for all models. The first is perplexity over the (unfiltered) test corpus of Gulordava et al. (2018). The second is accuracy on each of the 67 benchmarks in the BLiMP challenge set (Warstadt et al., 2020). Accuracy on the BLiMP benchmarks was assessed via the "full-sentence" method (Marvin and Linzen, 2018), where a "success", for any minimal pair, is defined by the model assigning a higher probability to the grammatical sentence in the minimal pair (s^+) than to the ungrammatical sentence (s^-). Sentence probabilities were extracted using the `minicons` library (Misra, 2022).

However, the FICT methodology's main advantage lies not in looking the performance of each model in isolation, but on the *difference* in performance between two models that are otherwise identical but for their training data. Thus, for each model and each BLiMP

benchmark, a change score, or *accuracy delta*, was calculated with respect to the average performance of all models of the same architecture trained on the FULL corpus (i.e. average over the five seeds).

To be more precise, with M a model type (i.e. $M \in \{\mathsf{LSTM}, \mathsf{Transformer}\}$), F a filter, and B a benchmark, $F(B)$ will refer to the filtered corpus targeting B, and M_F will refer to a model trained on F. We can then define the accuracy delta by:

$$\mathsf{acc}\Delta(M, F, B) := \mathsf{acc}_B^{M_F} - \overline{\mathsf{acc}_B^{M_{\text{FULL}}}}$$

where acc_B^M refers to the accuracy of model M on benchmark B. We will often be interested in the case where $F = F(B)$, i.e. the benchmark(s) corresponding to the corpus filter, but report others as well.

Chapter 4

RESULTS

4.1 Perplexity

4.1.1 Effect of Architecture

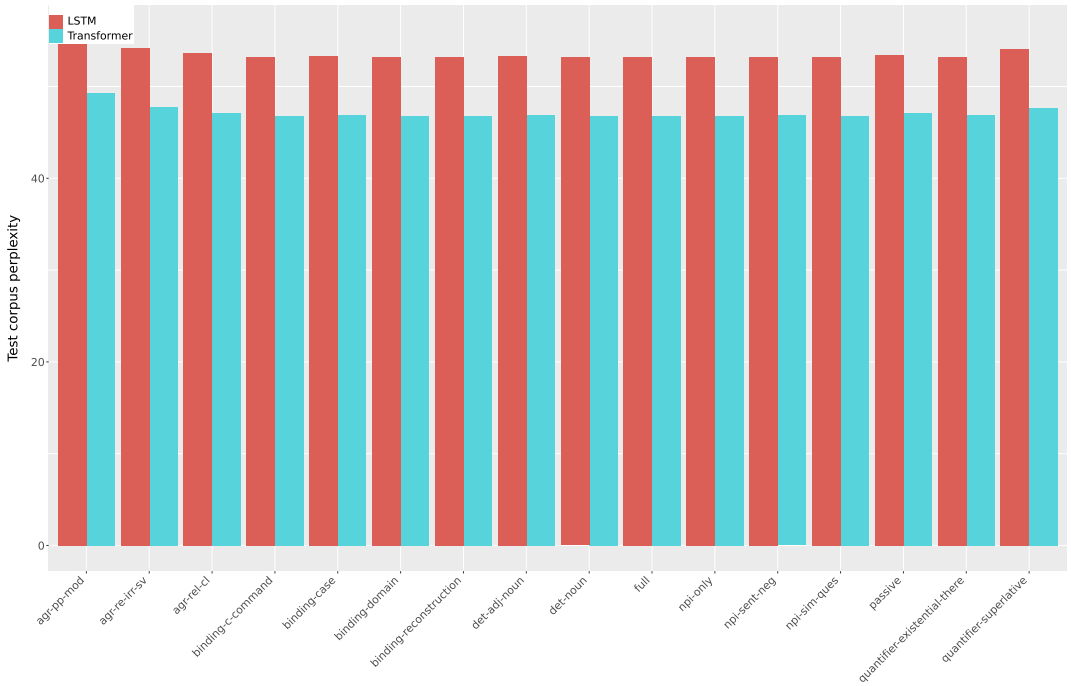

Figure 4.1: Mean test corpus perplexity by architecture for each training corpus.

As we can see in Figure 4.1, Transformers uniformly achieve lower perplexities on the test corpus than the LSTMs for all training corpora, as expected. The mean test perplexity across all corpora and random seeds was 47.13 for the Transformers and 53.56 for the LSTMs; a paired t-test of mean perplexities per corpus found the difference between the model types

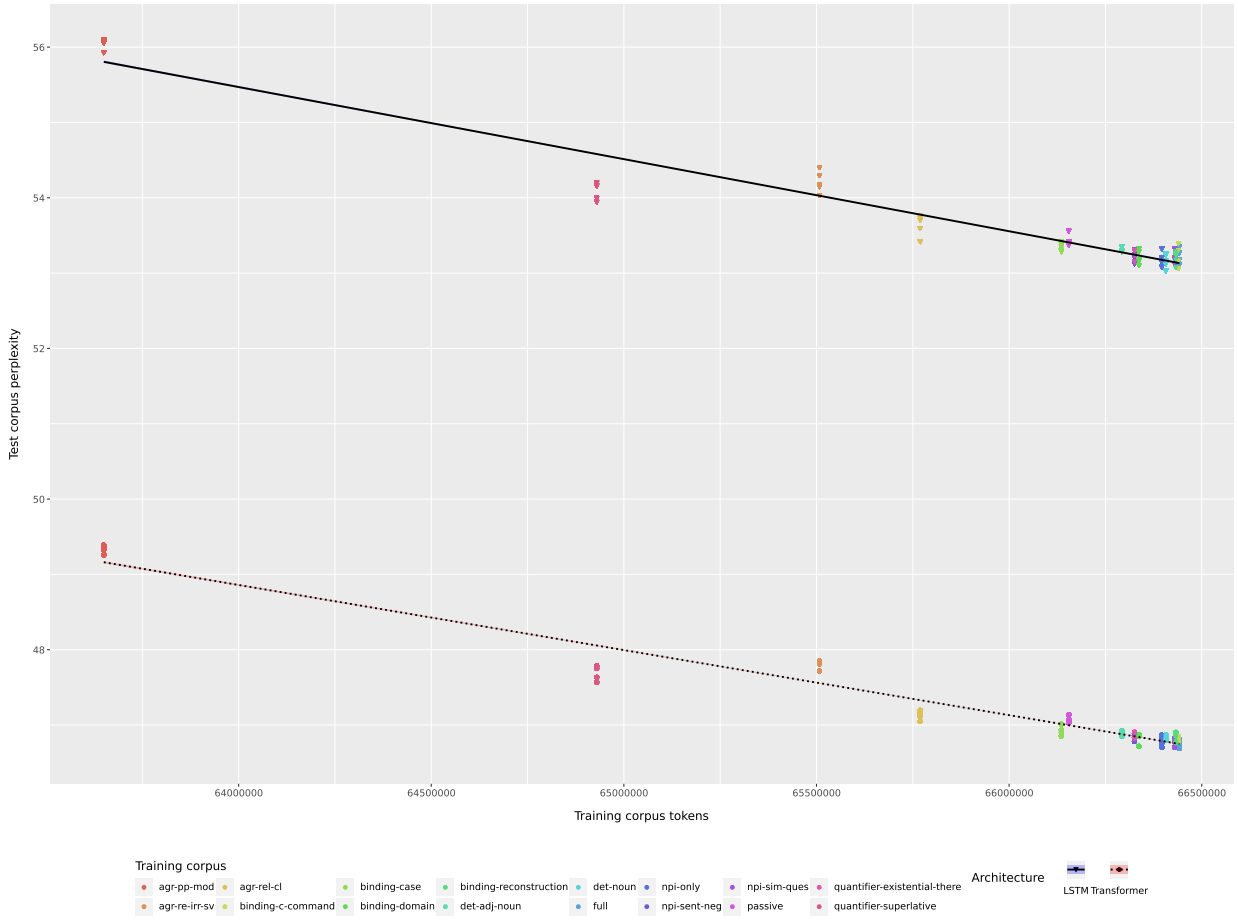

Figure 4.2: Test corpus perplexity as a function of the number of tokens in the training corpus. Solid/dashed lines correspond to the line of best fit for LSTMs and Transformers, respectively. The barely visible shaded area around each line of best fit is a 95% confidence interval.

to be significant ($t = 270.94$, $p \approx 0$). This result accords with the all prior literature on this subject: from the perspective of perplexity, Transformers are better language models than LSTMs.

4.1.2 Perplexity and Training Corpus Size

As noted in §3.2, while we downsampled all corpora to the same number of lines, the number of tokens varies between training corpora. Previous research has shown a clear negative

relationship between the number of tokens seen in training and test corpus perplexity. As seen in Figure 4.2, this effect is also present in our data, for both architectures (LSTM's: $r = -0.970$; Transformers: $r = -0.976$; pooled: $r = -0.202$, $\rho = -0.411$)[1].

4.2 BLiMP Accuracy

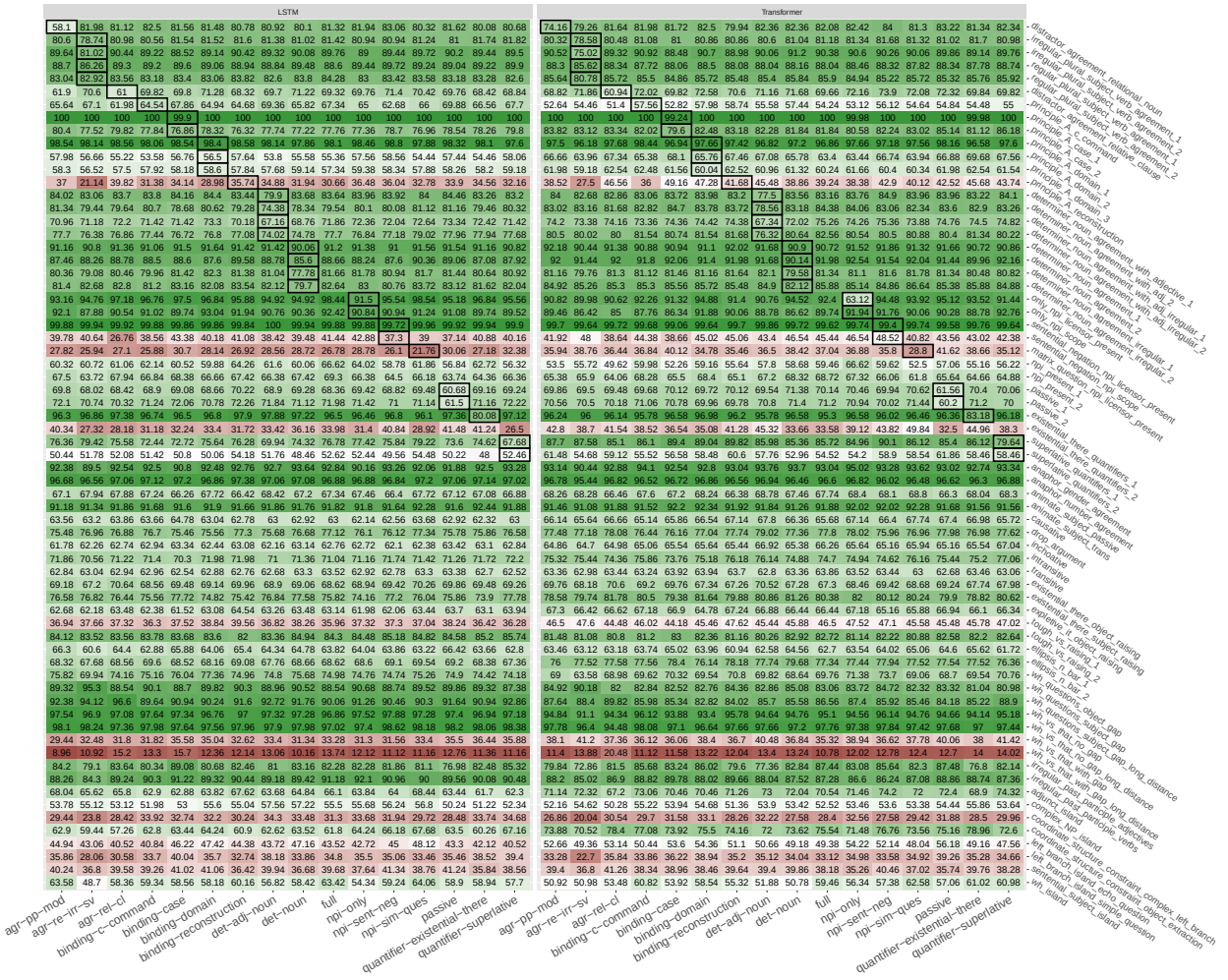

Figure 4.3: Raw accuracy scores on the BLiMP benchmarks, averaged across all models with the same architecture and training corpus. Boxes with bold outlines correspond to benchmarks targeted by the filter that produced the given model's training corpus(i.e. where $F = F(B)$).

[1]Where r refers to the Pearson correlation coefficient and ρ to Spearman's rank correlation coefficient.

In Figure 4.3, we report the average BLiMP accuracy of all models, averaged across the five starting seeds. Mean overall accuracy on all of BLiMP across different training corpora (i.e. $\overline{\mathsf{acc}}^{M_F}_{\text{ALL}}$) was 70.4 for the LSTMs and 71.9 for the Transformers. This result was statistically significant (paired $t = -17.38$, $p \approx 0$).

We next look only at benchmark accuracy data where the filtered corpus targeted a given benchmark, i.e. where $F = F_B$. Here, the mean is 68.8 for the Transformers and 66.7 for the LSTMs *and this difference is not statistically significant* (paired $t = -1.18$, $p = 0.258$). In other words, we find no difference in the two models' ability to make grammaticality judgments when trained on filtered data that forces them to perform subtle generalizations, despite differences in perplexity.

4.3 Accuracy Delta

In Figure 4.4, we report average accuracy deltas for all models (except FULL). In Figure 4.5, we plot these same values, along with 95% confidence intervals, aggregated factorially by architecture and whether the model's training corpus was generated by a filter targeting the specific benchmark indicated on the y-axis. Mean overall accuracy delta over all benchmarks and across all training corpora (i.e. $\overline{\Delta}(M, F, B)$) was -0.393 for the LSTMs and 0.0313 for the Transformers. This result was statistically significant (paired $t = -5.10$, $p \approx 0.00013$).

Focusing on the $F = F(B)$ cases in Figure 4.5 (i.e. the triangular points), we note that most values (and their corresponding confidence intervals) are negative, but generally cluster near zero, with a few outliers, such as the models trained on the EXISTENTIAL-THERE, AGR-PP-MOD, and NPI-ONLY corpora. These results suggest that, overall, learners *are* usually able to use various sorts of indirect evidence to acquire correct grammatical generalizations when direct evidence has been made unavailable, as otherwise we could expect much larger accuracy deltas across the board.

We may also observe that, for the cases where the absolute value of the accuracy deltas was appreciably larger than zero, it is not the case that one architecture is uniformly better than the other. For example, LSTMs perform better than Transformers (that is, their accu-

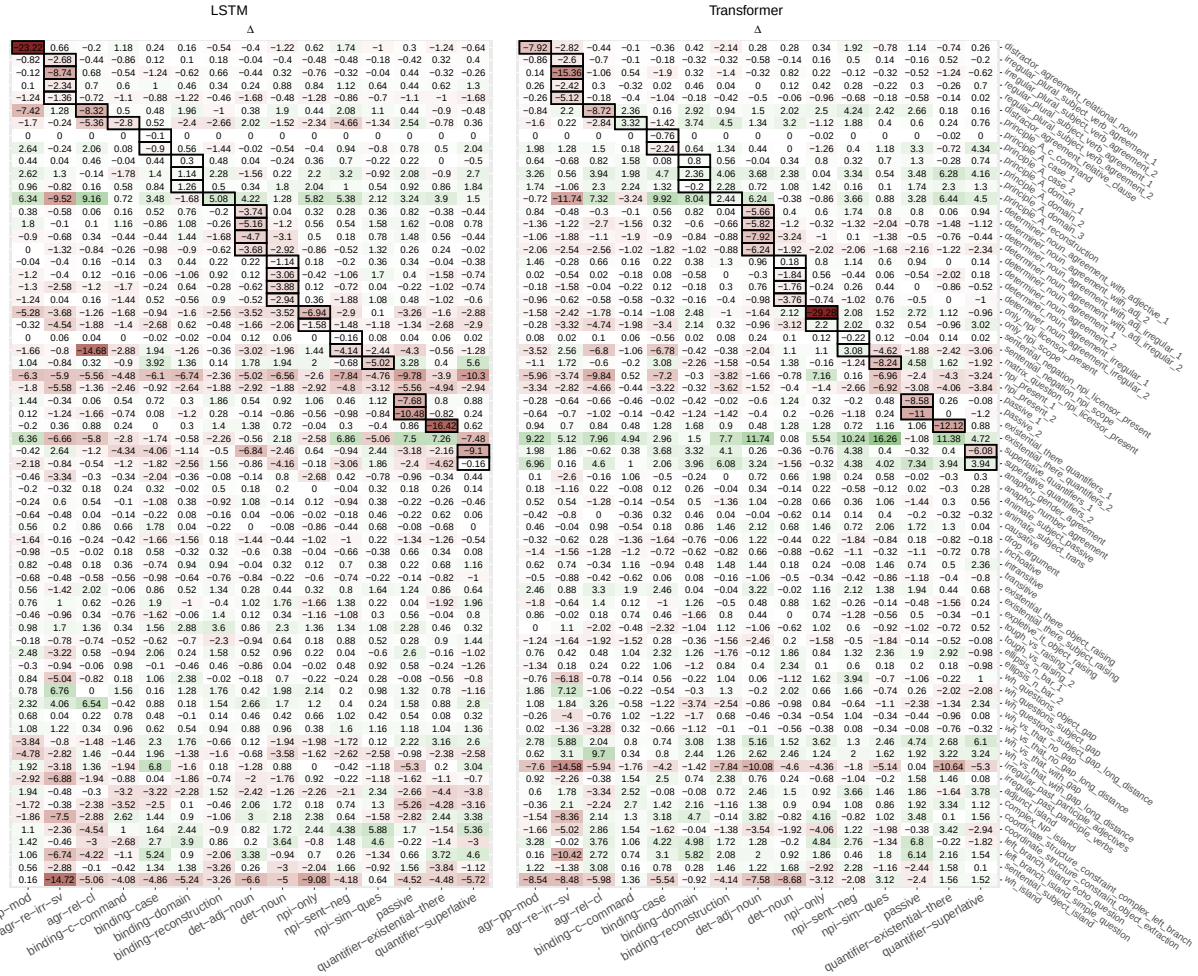

Figure 4.4: BLiMP benchmark accuracy delta ($\Delta(M, F, B)$) for the filtered corpora, averaged across seeds. Boxes with bold outlines correspond to benchmarks targeted by the model's corpus filter (i.e. where $F = F(B)$).

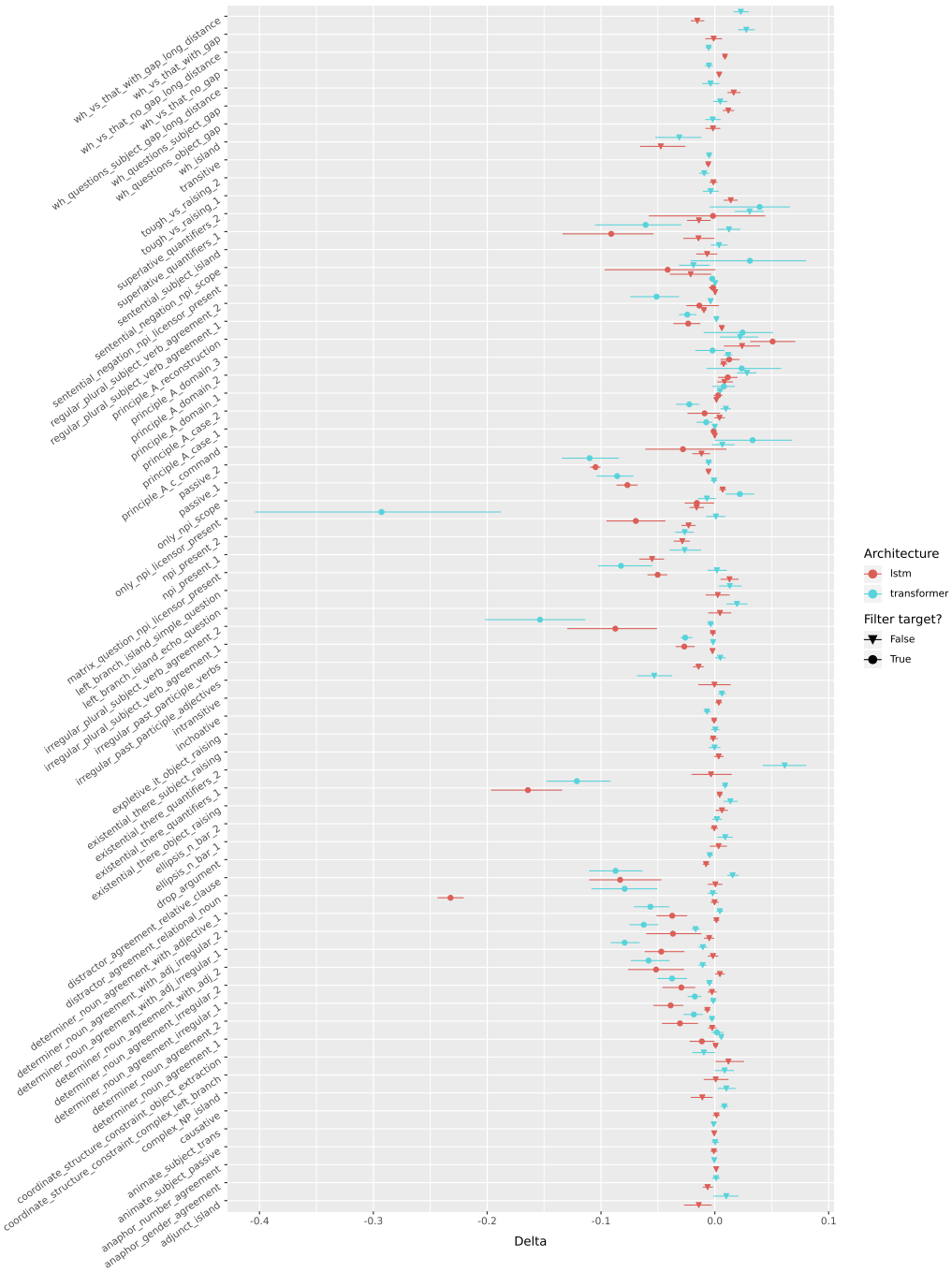

Figure 4.5: Accuracy delta for each model across all BLiMP benchmarks. Lines represent a bootstrapped 95% confidence interval about the mean. The shape of each point (triangle or circle) categorizes the models training on filtered corpora which targeted the grammatical phenomena measured by that BLiMP benchmark.

racy deltas are smaller in magnitude) on the benchmarks associated with the AGR-RE-IRR-SV and the NPI-ONLY corpora, while the converse is true for AGR-PP-MOD and QUANTIFIER-EXISTENTIAL-THERE. This is true *even* for phenomena that are seemingly relatively similar; for example, the AGR-PP-MOD and AGR-RE-IRR-SV-AGR filters are extremely similar, in that they both test long distance agreement in the present of a clausal distractor intervening between the subject and the verb; they differ only in the nature of that distractor. Yet, as noted, LSTMs trained on the AGR-RE-IRR-SV corpus have, on average, a less negative accuracy delta on the associated benchmarks than the analogous Transformer models ($\overline{\mathsf{acc}\Delta}(\text{LSTM}, \text{AGR-RE-IRR-SV}, F(B)) = -3.78$; for the Transformer, -6.38); conversely, on the models trained on the AGR-PP-MOD corpus, it is Transformers which have the smaller magnitude delta ($\overline{\mathsf{acc}\Delta}(\text{LSTM}, \text{AGR-PP-MOD}, F(B)) = -23.22$; Transformer, -7.92).

As in §4.2, we can make this precise by analyzing all of the accuracy deltas where $F = F_B$. The mean here is -5.41 for the LSTMs and -4.62 for the Transformers and this difference is not statistically significant (paired $t = -0.562$, $p = 0.583$). That means that we again find no difference between the two architectures in the extent to which filtering affects their accuracy, despite significant differences in perplexity. This suggests that perplexity *does not* predict the ability of a model to perform linguistic generalizations from indirect evidence.

4.3.1 Accuracy Delta and Training Corpus Size

In §4.1.2 we observed, on the basis of the data in Figure 4.2, a clear, negative relationship between the size of a model's training corpus (in tokens) and its test corpus perplexity. On the other hand, in Figure 4.6, we can see that the relationship between relationship between tokens seen in training data and accuracy delta is not nearly as straightforward (LSTM's: $r = 0.032$; Transformers: $r = 0.037$; pooled: $r \approx \rho = 0.034$). However, as seen in Figure 4.7, when we look only at the accuracy deltas of filtered corpus models on the benchmarks their filters target, this relationship re-emerges (LSTM's: $r = 0.482$; Transformers: $r = 0.079$; pooled: $r = 0.256, \rho = 0.294$), especially for LSTM's, though one much weaker than that between tokens seen in training and test corpus perplexity.

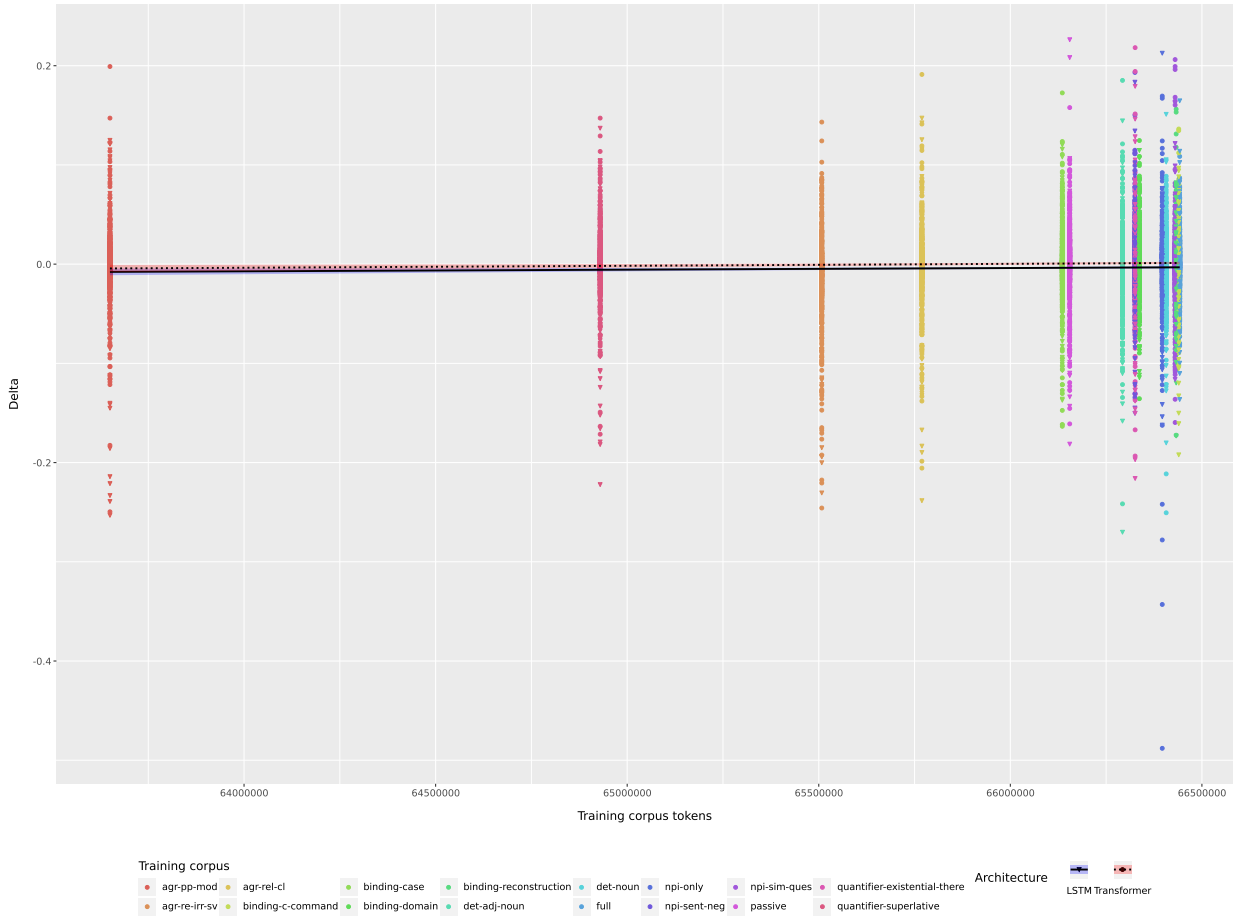

Figure 4.6: Accuracy delta as a function of the number of tokens in the training corpus, for all models. Solid/dashed lines correspond to the line of best fit for LSTMs and Transformers, respectively. The barely visible shaded area around each line of best fit is a 95% confidence interval.

4.4 Regression

In the final part of our analysis, we conducted an ordinary least squares regression analysis analysis to better understand the relationship between architecture and the models' ability to generalize to unseen contexts. Accuracy delta was the dependent variable, while the independent variables consisted of the variable of interest (architecture) and various control variables: the BLiMP benchmark for which the delta was computed, the training corpus used,

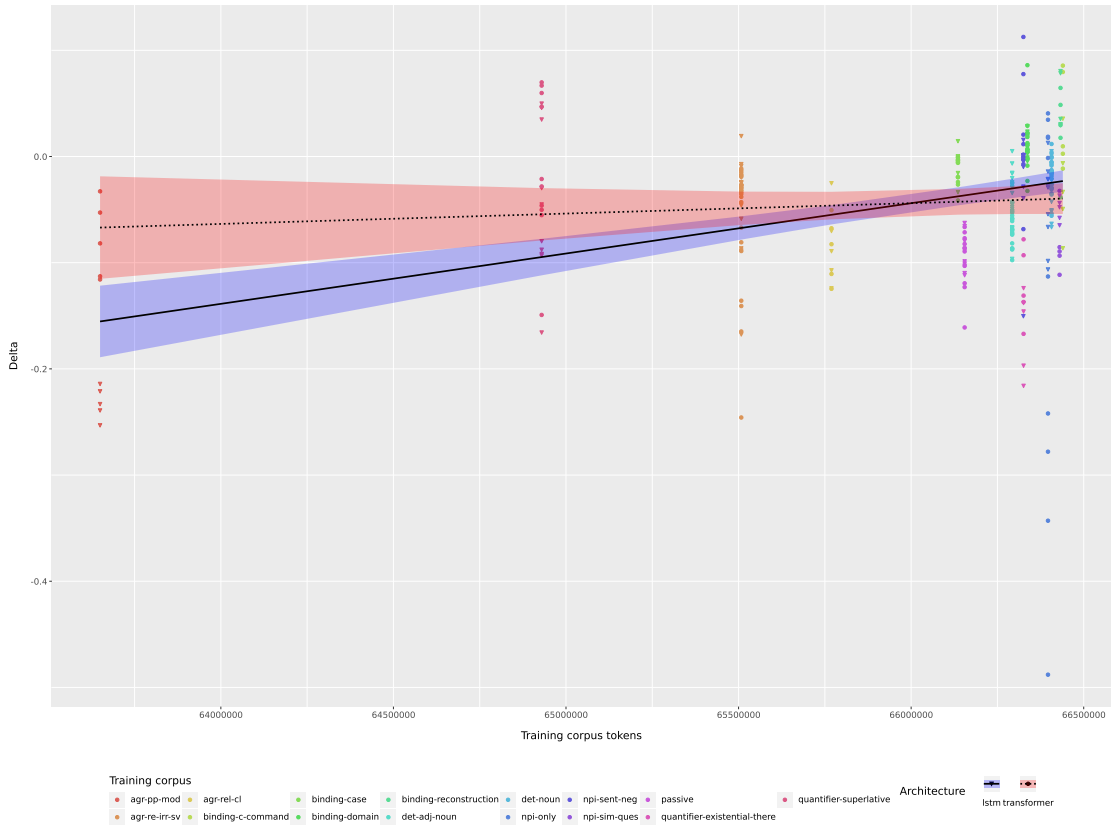

Figure 4.7: Training corpus tokens vs. accuracy delta for filtered corpora on the benchmarks they targeted, as a function of the number of tokens in the training corpus. Solid/dashed lines correspond to the line of best fit for LSTMs and Transformers, respectively, while the shaded areas represent a 95% confidence interval.

and whether the given benchmark was targeted by the given corpus' filter. As the dependent variable was a change score computed with respect to the FULL model's performance, this model was excluded from the regression. The results of this regression are reported in Table 4.1.

The t-test for the coefficient associated with architecture yielded a t-score of 6.295 ($p < 0.001$), indicating a statistically significant relationship, albeit an extremely small one ($\hat{\beta}_{\text{Transformer}} = 0.0045, 95\%\ \text{CI}[0.003, 0.006]$), in favor of Transformers.

Dep. Variable:	blimp_delta	**R-squared:**	0.152
Model:	OLS	**Adj. R-squared:**	0.145
Method:	Least Squares	**F-statistic:**	16.32
Date:	Sat, 02 Mar 2024	**Prob (F-statistic):**	2.51e−210
Time:	03:06:52	**Log-Likelihood:**	19191.
No. Observations:	10050	**AIC:**	−3.822e4
Df Residuals:	9967	**BIC:**	−3.762e4
Df Model:	82		
Covariance Type:	HC1		

	coef	std err	t	P> |t|	[0.025	0.975]
Intercept	-0.0050	0.005	-1.094	0.274	-0.014	0.004
corpus[T.binding-case]	0.0030	0.002	1.590	0.112	-0.001	0.007
corpus[T.binding-domain]	0.0054	0.002	3.072	0.002	0.002	0.009
corpus[T.binding-reconstruction]	0.0026	0.002	1.482	0.138	-0.001	0.006
corpus[T.det-adj-noun]	0.0011	0.002	0.590	0.555	-0.003	0.005
corpus[T.det-noun]	0.0011	0.002	0.622	0.534	-0.002	0.005
corpus[T.existential-there-quantifier]	-0.0013	0.002	-0.702	0.483	-0.005	0.002
corpus[T.npi-only]	0.0010	0.002	0.490	0.624	-0.003	0.005
corpus[T.npi-sent-neg]	0.0055	0.002	3.044	0.002	0.002	0.009
corpus[T.npi-sim-ques]	0.0023	0.002	1.263	0.207	-0.001	0.006
corpus[T.passive]	0.0021	0.002	1.126	0.260	-0.002	0.006
corpus[T.pp-mod-subj]	-0.0018	0.002	-0.895	0.371	-0.006	0.002
corpus[T.re-irr-sv-agr]	-0.0113	0.002	-5.209	0.000	-0.016	-0.007
corpus[T.rel-cl]	-0.0029	0.002	-1.479	0.139	-0.007	0.001
corpus[T.superlative-quantifier]	0.0021	0.002	1.096	0.273	-0.002	0.006
arch[T.transformer]	0.0045	0.001	6.295	0.000	0.003	0.006
filter_target[T.True]	-0.0432	0.004	-11.766	0.000	-0.050	-0.036
blimp_benchmark[T.anaphor_gender_agreement]	-0.0006	0.005	-0.135	0.893	-0.010	0.008
blimp_benchmark[T.anaphor_number_agreement]	0.0025	0.004	0.573	0.566	-0.006	0.011
blimp_benchmark[T.animate_subject_passive]	0.0019	0.005	0.416	0.678	-0.007	0.011
blimp_benchmark[T.animate_subject_trans]	0.0014	0.004	0.306	0.760	-0.007	0.010
blimp_benchmark[T.causative]	0.0074	0.005	1.630	0.103	-0.002	0.016
blimp_benchmark[T.complex_NP_island]	0.0017	0.006	0.305	0.761	-0.009	0.013
blimp_benchmark[T.coordinate_structure_constraint_complex_left_branch]	0.0072	0.006	1.251	0.211	-0.004	0.018
blimp_benchmark[T.coordinate_structure_constraint_object_extraction]	0.0034	0.006	0.552	0.581	-0.009	0.015
blimp_benchmark[T.determiner_noun_agreement_1]	0.0079	0.005	1.742	0.081	-0.001	0.017
blimp_benchmark[T.determiner_noun_agreement_2]	0.0010	0.005	0.226	0.821	-0.008	0.010
blimp_benchmark[T.determiner_noun_agreement_irregular_1]	-0.0008	0.005	-0.170	0.865	-0.010	0.008
blimp_benchmark[T.determiner_noun_agreement_irregular_2]	-0.0008	0.005	-0.175	0.861	-0.010	0.008
blimp_benchmark[T.determiner_noun_agreement_with_adj_2]	-0.0017	0.005	-0.367	0.714	-0.011	0.008
blimp_benchmark[T.determiner_noun_agreement_with_adj_irregular_1]	-0.0052	0.005	-1.106	0.269	-0.015	0.004
blimp_benchmark[T.determiner_noun_agreement_with_adj_irregular_2]	-0.0092	0.005	-1.942	0.052	-0.018	8.61e-05
blimp_benchmark[T.determiner_noun_agreement_with_adjective_1]	0.0048	0.005	1.070	0.285	-0.004	0.014
blimp_benchmark[T.distractor_agreement_relational_noun]	-0.0064	0.005	-1.190	0.234	-0.017	0.004
blimp_benchmark[T.distractor_agreement_relative_clause]	0.0074	0.005	1.475	0.140	-0.002	0.017
blimp_benchmark[T.drop_argument]	-0.0043	0.005	-0.938	0.348	-0.013	0.005
blimp_benchmark[T.ellipsis_n_bar_1]	0.0088	0.005	1.724	0.085	-0.001	0.019
blimp_benchmark[T.ellipsis_n_bar_2]	0.0029	0.005	0.626	0.531	-0.006	0.012
blimp_benchmark[T.existential_there_object_raising]	0.0128	0.005	2.622	0.009	0.003	0.022
blimp_benchmark[T.existential_there_quantifiers_1]	0.0022	0.005	0.443	0.658	-0.008	0.012
blimp_benchmark[T.existential_there_quantifiers_2]	0.0331	0.008	3.919	0.000	0.017	0.050
blimp_benchmark[T.existential_there_subject_raising]	0.0038	0.005	0.807	0.420	-0.005	0.013
blimp_benchmark[T.expletive_it_object_raising]	0.0016	0.005	0.356	0.722	-0.007	0.011
blimp_benchmark[T.inchoative]	-0.0018	0.005	-0.397	0.691	-0.011	0.007
blimp_benchmark[T.intransitive]	0.0074	0.005	1.613	0.107	-0.002	0.016
blimp_benchmark[T.irregular_past_participle_adjectives]	-0.0264	0.007	-3.563	0.000	-0.041	-0.012
blimp_benchmark[T.irregular_past_participle_verbs]	-0.0029	0.005	-0.615	0.538	-0.012	0.006
blimp_benchmark[T.irregular_plural_subject_verb_agreement_1]	0.0015	0.004	0.340	0.734	-0.007	0.010
blimp_benchmark[T.irregular_plural_subject_verb_agreement_2]	-0.0057	0.005	-1.168	0.243	-0.015	0.004
blimp_benchmark[T.left_branch_island_echo_question]	0.0149	0.006	2.578	0.010	0.004	0.026
blimp_benchmark[T.left_branch_island_simple_question]	0.0106	0.006	1.799	0.072	-0.001	0.022
blimp_benchmark[T.matrix_question_npi_licensor_present]	0.0080	0.005	1.494	0.135	-0.002	0.018
blimp_benchmark[T.npi_present_1]	-0.0414	0.006	-6.441	0.000	-0.054	-0.029
blimp_benchmark[T.npi_present_2]	-0.0272	0.005	-5.237	0.000	-0.037	-0.017
blimp_benchmark[T.only_npi_licensor_present]	-0.0181	0.006	-2.787	0.005	-0.031	-0.005
blimp_benchmark[T.only_npi_scope]	-0.0062	0.005	-1.194	0.233	-0.016	0.004
blimp_benchmark[T.passive_1]	0.0027	0.005	0.585	0.558	-0.006	0.012
blimp_benchmark[T.passive_2]	-0.0076	0.005	-1.629	0.103	-0.017	0.002
blimp_benchmark[T.principle_A_c_command]	0.0027	0.006	0.466	0.641	-0.009	0.014
blimp_benchmark[T.principle_A_case_1]	0.0047	0.004	1.056	0.291	-0.004	0.014
blimp_benchmark[T.principle_A_case_2]	0.0109	0.005	2.294	0.022	0.002	0.020
blimp_benchmark[T.principle_A_domain_1]	0.0084	0.005	1.856	0.063	-0.000	0.017
blimp_benchmark[T.principle_A_domain_2]	0.0246	0.005	4.618	0.000	0.014	0.035
blimp_benchmark[T.principle_A_domain_3]	0.0151	0.005	3.277	0.001	0.006	0.024
blimp_benchmark[T.principle_A_reconstruction]	0.0306	0.007	4.322	0.000	0.017	0.045
blimp_benchmark[T.regular_plural_subject_verb_agreement_1]	0.0072	0.005	1.607	0.108	-0.002	0.016
blimp_benchmark[T.regular_plural_subject_verb_agreement_2]	-0.0038	0.005	-0.830	0.406	-0.013	0.005
blimp_benchmark[T.sentential_negation_npi_licensor_present]	0.0052	0.004	1.161	0.246	-0.004	0.014
blimp_benchmark[T.sentential_negation_npi_scope]	-0.0153	0.007	-2.080	0.038	-0.030	-0.001
blimp_benchmark[T.sentential_subject_island]	0.0005	0.005	0.097	0.923	-0.010	0.011
blimp_benchmark[T.superlative_quantifiers_1]	-0.0010	0.006	-0.153	0.878	-0.013	0.011
blimp_benchmark[T.superlative_quantifiers_2]	0.0146	0.006	2.255	0.024	0.002	0.027
blimp_benchmark[T.tough_vs_raising_1]	0.0077	0.005	1.499	0.134	-0.002	0.018
blimp_benchmark[T.tough_vs_raising_2]	-0.0035	0.005	-0.741	0.459	-0.013	0.006
blimp_benchmark[T.transitive]	-0.0036	0.004	-0.804	0.421	-0.012	0.005
blimp_benchmark[T.wh_island]	-0.0398	0.008	-4.705	0.000	-0.056	-0.023
blimp_benchmark[T.wh_questions_object_gap]	0.0003	0.005	0.065	0.948	-0.009	0.010
blimp_benchmark[T.wh_questions_subject_gap]	0.0110	0.005	2.217	0.027	0.001	0.021
blimp_benchmark[T.wh_questions_subject_gap_long_distance]	0.0090	0.005	1.722	0.085	-0.001	0.019
blimp_benchmark[T.wh_vs_that_no_gap]	0.0015	0.005	0.324	0.746	-0.007	0.010
blimp_benchmark[T.wh_vs_that_no_gap_long_distance]	0.0040	0.005	0.884	0.377	-0.005	0.013
blimp_benchmark[T.wh_vs_that_with_gap]	0.0164	0.005	3.074	0.002	0.006	0.027
blimp_benchmark[T.wh_vs_that_with_gap_long_distance]	0.0061	0.005	1.168	0.243	-0.004	0.016

Omnibus:	1956.949	**Durbin-Watson:**	1.699
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	26064.213
Skew:	-0.548	**Prob(JB):**	0.00
Kurtosis:	10.813	**Cond. No.**	79.4

Table 4.1: Regression of delta on architecture plus various control variables. Standard Errors are heteroscedasticity robust (HC1).

Chapter 5

DISCUSSION

5.1 The Utility of Targeted Syntactic Evaluations

Our findings contribute to a growing body of research that suggest a dissociation between perplexity and affirm the utility of more targeted evaluations of linguistic competence in artificial learners. We contend that while perplexity can be useful as a general-purpose metric, it has numerous drawbacks illustrated in this investigation. Overall, our findings do not suggest any simple, straightforward, or general relationship between perplexity and performance on the BLiMP benchmark sets. As an example, consider our finding that Transformers consistently outperform LSTMs in terms of perplexity across all training corpora. This finding aligns with expectations, given the well-attested superior architectural capabilities of Transformers in capturing long-range dependencies and contextual information (Vaswani et al., 2017), and at first glance appears to be a rather uninteresting result. If we were to compare these two architectures solely on the basis of an information-theoretic measure like perplexity, we might prematurely conclude that Transformers have a marked advantage over LSTMs and are thus

However, the targeted linguistic evaluations we perform paint a much more nuanced picture. At best, Transformers appear to have only a very small advantage, which is frequently negated or even reversed in specific contexts. While our regression analysis found an extremely small effect on accuracy delta in favor of Transformers, it is evident from our findings that for a not insignificant number of *specific* linguistic phenomena, this advantage is negated or even reversed. The clearest example of this comes from the NPI-ONLY corpus: our Transformer models revealed a much greater sensitivity to ablations of this NPI-licensing environment from the training data compared to LSTMs. The case of the AGR-RE-IRR-SV

and AGR-PP-MOD corpora also provide a crucial example; despite both grammatical phenomena being extremely similar, LSTMs notably outperform Transformers on the former, while the opposite is true for the latter.

A naive reliance on simple information-theoretic measures would mask such differences. Thus, this research suggests that a key advantage of targeted linguistic evaluations over information-theoretic measures may lie in their greater *granularity* and *specificity*. That is, these evaluations allow us to locate and understand differences in language model performance that may be entirely opaque from the perspective of a broad information-theoretic measure like perplexity.

As discussed in chapter 4, we observe a clear negative relationship between training corpus size and perplexity, consistent with previous research. Interestingly, when considering only models trained on filtered corpora targeting specific benchmarks, we observe a re-emergence of the negative relationship between training corpus size and accuracy delta, particularly for LSTM models. This provides further evidence that while larger training corpora may lead to lower perplexity, they do not necessarily translate into improved performance on targeted linguistic evaluations. Instead, the effectiveness of training data appears to depend on its relevance to the linguistic phenomena being evaluated, or, stated more generally, on its overall quality.

In a carefully controlled setting and for a wide range of phenomena, we have demonstrated that the training objective of minimizing perplexity does not robustly predict linguistic generalization. This raises interesting questions on the relation between perplexity and grammaticality judgments (Lau et al., 2017): while Transformers are better at *memorizing* the structure of its training data, we show they are less capable than LSTMs of forming robust linguistic generalizations. An interesting step for future work would be to uncover what language modeling aspects Transformers *do* excel at, which allows them to obtain a superior test perplexity.

5.2 Insights into Human Language Acquisition

Our study also builds on the insights of Warstadt (2022) in the use of artificial learners as models for understanding human language acquisition. As previously discussed, Warstadt (2022) conducted a "proof-of-concept... large-scale controlled ablation study on the input to model learners," and found that direct attestation of linguistic evidence is not strictly necessary for the development of sophisticated linguistic generalizations. Rather, learners can leverage much more indirect sources of evidence to arrive at the correct generalizations.

Where earlier work has focused on specific linguistic constructions, such as subject auxiliary inversion (Warstadt, 2022), relative clauses (Prasad et al., 2019), negative polarity items (Jumelet et al., 2021; Weber et al., 2021), and rare adjective-noun constructions (Misra and Mahowald, 2024) the results of this thesis essentially confirm a similar result for a much wider array of syntactic and semantic phenomena. While in many cases the ablations we performed did clearly negatively affect the performance of our artificial learners on the relevant linguistic evaluations, the magnitude of this effect was generally quite small for all but a small handful of the linguistic phenomena we analyzed. In general, even when tested on the specific benchmarks corresponding to the environments that were ablated from their input, models still perform considerably better than chance. Thus, our research provides evidence in favor of Warstadt's (2022) indirect evidence hypothesis.

Notably, we find that this is true not only for filters where there are fairly obvious sources of indirect evidence (as enumerated in §3.1), but also for filters where potential sources of indirect evidence for a correct generalization are much less clear (such as the QUANTIFIER-SUPERLATIVE filter). This suggests that there may be complex, poorly understand mechanisms by which certain linguistic generalizations can be derived via highly indirect means. Thus, our results open a door to future research that can provide a more thorough account of the source of these generalizations, with potentially significant ramifications for linguistic science.

5.3 Implications and Future Research

As just discussed, the primary contribution of this thesis has been the development of the FiCT method and the use of it to demonstrate LMs' successful generalization from indirect evidence across a *wide range* of linguistic phenomena. This success raises a very natural follow-up question: what explains this successful generalization behavior?

While a complete answer to this question must await future work, a detailed look at the NPI cases can provide insight into what an answer may look like. Jumelet et al. (2021) used a filtered corpus method to test LSTM LMs' understanding of negative polarity items, but then also did a further analysis to examine the basis upon which the models made their grammaticality judgments. In particular, they found (via probing classifiers) that LMs' were successfully recognizing the *monotonicity* of a linguistic environment and (via a novel correlation method) that these judgments of monotonicity were highly correlated with the LMs' judgment of NPI acceptability, reflecting human acceptability judgments (Denić et al., 2021; Chemla et al., 2011).

This example suggests two paths forward for explaining the generalization observations in the present paper. On the one hand, in the same way that the monotonicity explanation was inspired by human generalization, detailed explanations of individual cases of generalization can be developed with human behavior as an initial inspiration. On the other hand, in the same way that this paper extends the filtered corpus training method to a much wider range of phenomena, one can attempt to generalize these forms of explanation on the breadth axis as well.

Chapter 6

CONCLUSION

We introduced the **Fi**ltered **C**orpus **T**raining methodology and applied it to a wide range of linguistic constructions from the BLiMP benchmark. Our results show that while Transformers are better language models (as measured via perplexity) than comparable LSTMs, the latter generalize equally well (as measured via $\mathsf{acc}\Delta$). While the regression analysis we conducted did find a statistically significant advantage in favor of the Transformer models, this effect was overall extremely small. Importantly, a fine-grained analysis of benchmark-level results shows that any architectural advantage on one grammatical phenomenon does not necessarily translate to an advantage on other phenomena, even closely related ones. This contrasts starkly with the uniformly superior performance of Transformers over LSTMs when viewed solely through the lens of perplexity, across all training corpora. Furthermore, while training corpus token count is a known to be negatively correlated with perplexity (a finding we replicate with our models), it bears no relationship with $\mathsf{acc}\Delta$, suggesting that sheer quantity cannot substitute for low quality data.

We have also shown that all of our LMs exhibit a strong ability to generalize from indirect evidence, as demonstrated by the relatively low $\mathsf{acc}\Delta$ scores in general. This is true despite the relatively low parameter count of our models and small size of their training data. This has potentially important implications for discussions surrounding the poverty of the stimulus debate, as it suggests that human learners may also be able to make generalizations from apparently impoverished data.

Future work may build on this thesis in a number of ways. First, it may extend this approach to models of different sizes and pretraining corpora and analyze other forms of lexical and structural generalization through the lens of filtered corpus training. Second, it remains

an open question how or why Transformers are able to achieve lower perplexities on training data if such performance does not necessarily translate in the domain of targeted syntactic evaluation. Furthermore, while we have shown that models *are* capable of generalizing from indirect evidence in the first place, we have only suggested a number of potential sources of such generalization ability; it remains to be shown that the actual linguistic and architectural bases of such abilities lie. An auxiliary question would be that of determining why apparent architectural advantages on one type of linguistic phenomena do not necessarily translate to advantages on very closely related phenomena. We leave these exciting pursuits to future work.

BIBLIOGRAPHY

Emmanuel Chemla, Vincent Homer, and Daniel Rothschild. Modularity and intuitions in formal semantics: The case of polarity items. *Linguistics and Philosophy*, 34(6):537–570, 2011. ISSN 01650157. doi: 10.1007/s10988-012-9106-0.

Noam Chomsky. *Syntactic Structures*. De Gruyter Mouton, Boston, 1957. ISBN 9783112316009. doi: doi:10.1515/9783112316009. URL https://doi.org/10.1515/9783112316009.

Noam Chomsky. *Lectures on Government and Binding*. De Gruyter Mouton, Berlin, New York, 1993. ISBN 9783110884166. doi: doi:10.1515/9783110884166. URL https://doi.org/10.1515/9783110884166.

Milica Denić, Vincent Homer, Daniel Rothschild, and Emmanuel Chemla. The influence of polarity items on inferential judgments. *Cognition*, 215:104791, October 2021. ISSN 0010-0277. doi: 10.1016/j.cognition.2021.104791.

Victoria Fossum and Roger Levy. Sequential vs. hierarchical syntactic models of human incremental sentence processing. In David Reitter and Roger Levy, editors, *Proceedings of the 3rd Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2012)*, pages 61–69, Montréal, Canada, June 2012. Association for Computational Linguistics. URL https://aclanthology.org/W12-1706.

Stefan L. Frank and Rens Bod. Insensitivity of the human sentence-processing system to hierarchical structure. *Psychological Science*, 22(6):829–834, 2011. ISSN 09567976, 14679280. URL http://www.jstor.org/stable/25835458.

Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger

Levy. Neural language models as psycholinguistic subjects: Representations of syntactic state. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 32–42, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1004. URL https://aclanthology.org/N19-1004.

Adam Goodkind and Klinton Bicknell. Predictive power of word surprisal for reading times is a linear function of language model quality. In Asad Sayeed, Cassandra Jacobs, Tal Linzen, and Marten van Schijndel, editors, *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 10–18, Salt Lake City, Utah, January 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-0102. URL https://aclanthology.org/W18-0102.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless green recurrent networks dream hierarchically. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1108. URL https://aclanthology.org/N18-1108.

Yiding Hao, Simon Mendelsohn, Rachel Sterneck, Randi Martinez, and Robert Frank. Probabilistic predictions of people perusing: Evaluating metrics of language model performance for psycholinguistic modeling. In Emmanuele Chersoni, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus, editors, *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 75–86, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.cmcl-1.10. URL https://aclanthology.org/2020.cmcl-1.10.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. A systematic assessment of syntactic generalization in neural language models. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.158. URL https://aclanthology.org/2020.acl-main.158.

Dieuwke Hupkes and Willem Zuidema. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure (extended abstract). In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 5617–5621. International Joint Conferences on Artificial Intelligence Organization, 7 2018. doi: 10.24963/ijcai.2018/796. URL https://doi.org/10.24963/ijcai.2018/796.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, Dennis Ulmer, Florian Schottmann, Khuyagbaatar Batsuren, Kaiser Sun, Koustuv Sinha, Leila Khalatbari, Maria Ryskina, Rita Frieske, Ryan Cotterell, and Zhijing Jin. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5:1161–1174, 10 2023. doi: 10.1038/s42256-023-00729-y.

Jaap Jumelet and Dieuwke Hupkes. Do language models understand anything? on the ability of LSTMs to understand negative polarity items. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi, editors, *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 222–231, Brussels, Belgium,

November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5424. URL https://aclanthology.org/W18-5424.

Jaap Jumelet, Milica Denic, Jakub Szymanik, Dieuwke Hupkes, and Shane Steinert-Threlkeld. Language models use monotonicity to assess NPI licensing. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4958–4969, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.439. URL https://aclanthology.org/2021.findings-acl.439.

Daniel Jurafsky and James H. Martin. *Speech and language processing*. Upper Saddle River, 3rd edition, 2024. URL https://web.stanford.edu/~jurafsky/slp3/. Unpublished draft.

Najoung Kim and Tal Linzen. COGS: A compositional generalization challenge based on semantic interpretation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9087–9105, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.731. URL https://aclanthology.org/2020.emnlp-main.731.

Tatsuki Kuribayashi, Yohei Oseki, Takumi Ito, Ryo Yoshida, Masayuki Asahara, and Kentaro Inui. Lower perplexity is not always human-like. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5203–5217, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.405. URL https://aclanthology.org/2021.acl-long.405.

Jey Han Lau, Alexander Clark, and Shalom Lappin. Grammaticality, acceptability, and

probability: A probabilistic view of linguistic knowledge. *Cognitive science*, 41 5:1202–1241, 2017. URL https://api.semanticscholar.org/CorpusID:1056628.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535, 2016. doi: 10.1162/tacl_a_00115. URL https://aclanthology.org/Q16-1037.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. URL https://api.semanticscholar.org/CorpusID:53592270.

Rebecca Marvin and Tal Linzen. Targeted syntactic evaluation of language models. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1151. URL https://aclanthology.org/D18-1151.

Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334.

Gary Milsark. *Existential Sentences in English*. PhD thesis, MIT, Cambridge, MA, 1974.

Kanishka Misra. minicons: Enabling Flexible Behavioral and Representational Analyses of Transformer Language Models. *arXiv preprint arXiv:2203.13112*, 2022. URL https://arxiv.org/abs/2203.13112.

Kanishka Misra and Kyle Mahowald. Language Models Learn Rare Phenomena from Less Rare Phenomena: The Case of the Missing AANNs, 2024. URL https://arxiv.org/abs/2403.19827.

Joakim Nivre, Daniel Zeman, Filip Ginter, and Francis Tyers. Universal Dependencies. In Alexandre Klementiev and Lucia Specia, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts*, Valencia, Spain, April 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-5001.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. Universal Dependencies v2: An evergrowing multilingual treebank collection. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.497.

Byung-Doh Oh and William Schuler. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350, 2023a. doi: 10.1162/tacl_a_00548. URL https://aclanthology.org/2023.tacl-1.20.

Byung-Doh Oh and William Schuler. Transformer-based language model surprisal predicts human reading times best with about two billion training tokens. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1915–1921, Singapore, December 2023b. As-

sociation for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.128. URL https://aclanthology.org/2023.findings-emnlp.128.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Abhinav Patil, Jaap Jumelet, Yu Ying Chiu, Andy Lapastora, Peter Shen, Lexie Wang, Clevis Willrich, and Shane Steinert-Threlkeld. Filtered Corpus Training (FiCT) Shows that Language Models can Generalize from Indirect Evidence, 2024. URL https://arxiv.org/abs/2405.15750.

Grusha Prasad, Marten van Schijndel, and Tal Linzen. Using priming to uncover the organization of syntactic representations in neural language models. In Mohit Bansal and Aline Villavicencio, editors, *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 66–76, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1007. URL https://aclanthology.org/K19-1007.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020. URL https://nlp.stanford.edu/pubs/qi2020stanza.pdf.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Lan-

guage models are unsupervised multitask learners. 2019. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. Masked language model scoring. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.240. URL https://aclanthology.org/2020.acl-main.240.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Alex Warstadt. *Artificial Neural Networks as Models of Human Language Acquisition*. PhD thesis, New York University, 2022. URL https://www.proquest.com/dissertations-theses/artificial-neural-networks-as-models-human/docview/2735573851/se-2.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020. doi: 10.1162/tacl_a_00321. URL https://aclanthology.org/2020.tacl-1.25.

Lucas Weber, Jaap Jumelet, Elia Bruni, and Dieuwke Hupkes. Language modelling as a multi-task problem. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2049–2060, Online, April 2021. As-

sociation for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.176. URL https://aclanthology.org/2021.eacl-main.176.

Lucas Weber, Jaap Jumelet, Elia Bruni, and Dieuwke Hupkes. Interpretability of language models via task spaces. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand, July 2024. Association for Computational Linguistics.

Jason Wei, Dan Garrette, Tal Linzen, and Ellie Pavlick. Frequency effects on syntactic rule learning in transformers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 932–948, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.72. URL https://aclanthology.org/2021.emnlp-main.72.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. What do RNN language models learn about filler–gap dependencies? In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi, editors, *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5423. URL https://aclanthology.org/W18-5423.

Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and Roger Philip Levy. On the predictive power of neural language models for human real-time comprehension behavior. *ArXiv*, abs/2006.01912, 2020. URL https://api.semanticscholar.org/CorpusID:219261016.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush.

Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer language models, 2022.

Appendix A

TRAINING HYPERPARAMETERS

adam_beta1	0.9
adam_beta2	0.999
adam_epsilon	1e-08
dataloader_num_workers	8
evaluation_strategy	epoch
fp16	True
gradient_accumulation_steps	1
ignore_data_skip	True
learning_rate	5e-05
lr_scheduler_type	linear
num_train_epochs	40
per_device_train_batch_size	32
per_device_eval_batch_size	32
optim	adamw_torch
seed	0,1,2,3,4
save_strategy	epoch

Table A.1: Selected training hyperparameters, as provided to the `transformers` package's `TrainingArguments` class. Any omitted values were set to the defaults associated with version 4.30.2 of the `transformers` package.